

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 07-084804

(43)Date of publication of application : 31.03.1995

(51)Int.Cl.

G06F 9/46

(21)Application number : 05-232687

(71)Applicant : FUJITSU LTD

(22)Date of filing : 20.09.1993

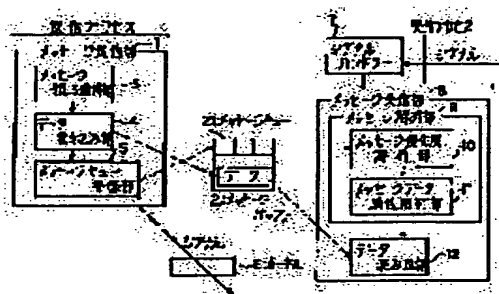
(72)Inventor : KAMIKURA KIYOUKO

(54) INTER-PROCESS COMMUNICATION SYSTEM

(57)Abstract:

PURPOSE: To preferentially process data which are generated in a communication from a process to another process and need to be processed preferentially as to the inter-process communication system which performs a communication between processes.

CONSTITUTION: This system is equipped with a transmitting process provided with a message area acquisition part 3 which acquires an area in a message buffer 2, a data write part 4 which writes data in the acquired area, and a message queue registration part 5 which registers a message whose priority level is set in a message queue 21 and a receiving process provided with a message priority analysis part 10 which finds a message having high priority among messages registered in the message queue 21 in response to the reception of a signal and a data read part 12 which reads the data corresponding to the found message with the high priority out of the message buffer 2.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平7-84804

(43)公開日 平成7年(1995)3月31日

(51)Int.Cl.⁶

G 0 6 F 9/46

識別記号

庁内整理番号

3 4 0 B 8120-5B

F I

技術表示箇所

審査請求 未請求 請求項の数4 O L (全 11 頁)

(21)出願番号 特願平5-232687

(22)出願日 平成5年(1993)9月20日

(71)出願人 000005223

富士通株式会社

神奈川県川崎市中原区上小田中1015番地

(72)発明者 神倉 鏡子

神奈川県川崎市中原区上小田中1015番地

富士通株式会社内

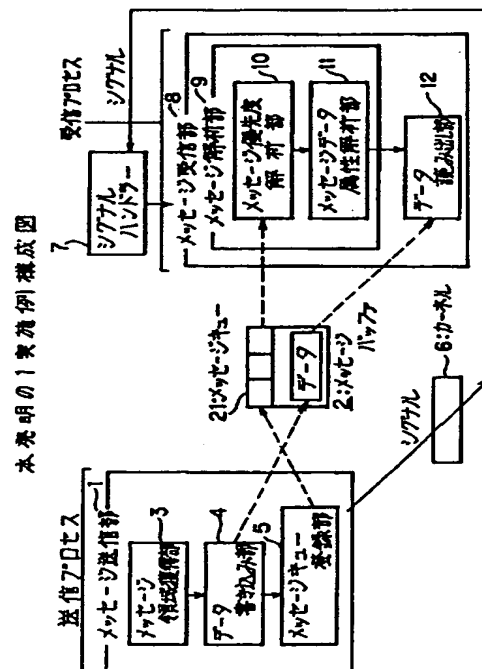
(74)代理人 弁理士 岡田 守弘

(54)【発明の名称】 プロセス間通信方式

(57)【要約】

【目的】 本発明は、プロセス間で通信を行なうプロセス間通信方式に関し、プロセスから他のプロセスに通信が行われている中で発生した優先処理の必要なデータを優先的に通信することを目的とする。

【構成】 メッセージバッファ2から領域を獲得するメッセージ領域獲得部3と、獲得した領域にデータを書き込むデータ書き込み部4と、優先順位を設定したメッセージをメッセージキュー21に登録するメッセージキュー登録部5とを設けた送信プロセスと、シグナルを受信したことに対応して、メッセージキュー21に登録されているメッセージの中から優先度の高いメッセージを見つけるメッセージ優先度解析部10と、見つけた優先度の高いメッセージに対応するデータをメッセージバッファ2の領域から読み出すデータ読み出し部12とを設けた受信プロセスを備えるように構成する。



1

【特許請求の範囲】

【請求項1】プロセス間で通信を行なうプロセス間通信方式において、

メッセージバッファ（2）から領域を獲得するメッセージ領域獲得部（3）と、

この獲得した領域に、送信しようとするデータを書き込むデータ書き込み部（4）と、

データに関する優先順位を設定したメッセージをメッセージキュー（21）に登録するメッセージキュー登録部（5）とを設けた送信プロセスと、

この送信プロセスが送信したシグナルを受信したことに対応して、メッセージキュー（21）に登録されているメッセージの中から優先度の高いメッセージ（あるいは複数の優先度の高いメッセージがあったときは最も早く登録されたメッセージ）を見つけるメッセージ優先度解析部（10）と、

この見つけた優先度の高いメッセージに対応するデータを上記メッセージバッファ（2）の領域から読み出すデータ読み出し部（12）とを設けた受信プロセスを備えたことを特徴とするプロセス間通信方式。

【請求項2】プロセス間で通信を行なうプロセス間通信方式において、

データおよび書き込み済を表す制御フラグを書き込む専用バッファ（19）と、

この専用バッファ（19）に、送信しようとするデータを書き込みおよび上記制御フラグを書き込み済にセットするデータ書き込み部（15）とを設けた送信プロセスと、

上記専用バッファ（19）中の制御フラグが書き込み済かを所定時間毎に判別し、書き込み済の場合にデータを読み出すおよび制御フラグをリセットするデータ読み出し部（17）とを設けた受信プロセスを備えたことを特徴とするプロセス間通信方式。

【請求項3】プロセス間で通信を行なうプロセス間通信方式において、

データを書き込む専用バッファ（19）と、

この専用バッファ（19）に、送信しようとするデータを書き込むデータ書き込み部（25）とを設けた送信プロセスと、

この送信プロセスが送信した特別シグナルを受信したことに対応して、上記専用バッファ（19）からデータを読み出すデータ読み出し部（32）とを設けた受信プロセスを備えたことを特徴とするプロセス間通信方式。

【請求項4】請求項1および請求項3の両者を備えたことを特徴とするプロセス間通信方式。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、プロセス間で通信を行なうプロセス間通信方式であって、計算機におけるマルチプロセスOS上のプロセス間通信機能を利用したプロ

2

セス間通信方式に関するものである。

【0002】

【従来の技術】マルチプロセッサOSの環境下で、並行して実行する複数のプロセス間でデータのやりとりを行なう方法として、プロセス間通信が用意されている。プロセス間通信機能として、シグナル、メッセージ、共有メモリ、セマフォなどがある。複数のプロセス間で自由に不特定サイズの多数のデータを送受信するには、通常、図10の（a）に示すメッセージを用いてデータのやり取りを行なう。簡単に説明すると、

（1） プロセスAがメッセージバッファ41から獲得した領域にデータを書き込むと共にメッセージをメッセージキュー42に登録した後、シグナルを受信側のプロセスBに送る。

【0003】（2） シグナルを受信したプロセスBは、メッセージキュー42からメッセージを取り出してこれをもとにメッセージバッファ41からデータを読み出す。これにより、プロセスAからプロセスBにデータが渡されたこととなる。

【0004】図10の（b）は、全てのプロセスからアクセス可能な1つのメッセージバッファを用意した例を示す。動作を説明する。

（1） アプリケーションに関係する全てのプロセスからアクセス可能なメッセージバッファ41を1つ用意する。通信を行なう場合、メッセージバッファ41から必要なサイズ分の空領域を獲得し、送信側のプロセスがこの空領域にデータを書き込むと共にメッセージをメッセージキュー42に登録した後、シグナルを受信側のプロセスに送信する。

【0005】（2） シグナルを受信した受信側のプロセスは、メッセージキュー42からメッセージを取り出してこれをもとにメッセージバッファ41からデータを読み出す。

【0006】図10の（c）は、プロセスとプロセスが1対1に対応するメッセージバッファ41を用意した例を示す。動作を説明する。

（1） アプリケーションに関係する全てのプロセスが、プロセスと1対1に対応するメッセージバッファ41をそれぞれ用意する。通信を行なう場合、送信側のプロセスが受信側のプロセスのメッセージバッファ41の空領域を獲得し、これにデータを書き込むと共にメッセージをメッセージキュー42に登録した後、シグナルを受信側のプロセスに送信する。

【0007】（2） シグナルを受信した受信側のプロセスは、メッセージキュー42からメッセージを順次取り出してこれの順にメッセージバッファ41からデータを読み出す。

【0008】

【発明が解決しようとする課題】上述した従来のプロセス間通信によれば、優先的に処理したい通信が発生して

3

も優先的に通信処理を行なうことができないという問題があった。例えばペンコンピュータで、電子ペンを用いてディスプレイ上に描画するような場合、ペンの座標データを最優先で処理しなければならない。もし少しでも通信に遅延が発生すると、ペンの数ミリ後を描画するような事態が起こる。これでは、ペンのヒューマンインタフェース上、使い難く大きな問題となる。

【0009】本発明は、これらの問題を解決するため、プロセスから他のプロセスに通信が行われている中で発生した優先処理に必要なデータを優先的に通信することを目的としている。

【0010】

【課題を解決するための手段】図1、図3および図5を参照して課題を解決するための手段を説明する。図1、図3および図5において、メッセージバッファ2は、データを書き込むものである。メッセージキュー21は、優先順位を高く設定などしたメッセージを登録するものである。

【0011】メッセージ領域獲得部3は、メッセージバッファ2からデータを格納する領域を獲得するものである。データ書き込み部4は、メッセージバッファ2中から獲得した領域にデータを書き込むものである。

【0012】メッセージキュー登録部5は、優先順位を高く設定などしたメッセージをメッセージキュー21に登録するものである。メッセージ優先度解析部10は、メッセージキュー21に登録されているメッセージ中から優先度の高いメッセージ（あるいは複数の優先度の高いメッセージがあったときは最も早く登録されたメッセージ）を見つけるものである。

【0013】データ読み出し部12は、メッセージバッファ2の領域からデータを読み出すものである。データ書き込み部15は、専用バッファ19に、送信しようとするデータを書き込みおよび制御フラグを書き込み済にセットするものである。

【0014】データ読み出し部17は、専用バッファ19中の制御フラグが書き込み済かを所定時間毎に判別し、書き込み済の場合にデータを読み出すおよび制御フラグをクリアするものである。

【0015】専用バッファ19は、データおよび書き込み済を表す制御フラグなどを書き込まれるものである。データ書き込み部25は、専用バッファ19に、送信しようとするデータを書き込むものである。

【0016】データ読み出し部32は、専用バッファ19からデータを読み出すものである。

【0017】

【作用】本発明は、図1に示すように、送信プロセスを構成するメッセージ領域獲得部3がメッセージバッファ2から領域を獲得し、データ書き込み部4が獲得した領域に送信しようとするデータを書き込み、メッセージキュー登録部5がデータに関する優先順位、データのアド

4

レスなどを設定したメッセージをメッセージキュー21に登録した後、シグナルを受信プロセスに送信し、このシグナルを受信した受信プロセスを構成するメッセージ優先度解析部10がメッセージキュー21に登録されているメッセージ中から優先度の高いメッセージ（あるいは複数の優先度の高いメッセージがあったときは最も早く登録されたメッセージ）を見つけ、メッセージデータ属性解析部11がメッセージを解析し、データのアドレスをデータ読み出し部12に渡し、メッセージバッファ2の領域から読み出し、プロセス間通信を行なうようにしている。

【0018】また、図3に示すように、送信プロセスを構成するデータ書き込み部15が専用バッファ19に送信しようとするデータを書き込みおよび制御フラグを書き込み済にセットし、受信プロセスを構成するデータ読み出し部17が専用バッファ19中の制御フラグが書き込み済かを所定時間毎に判別し、書き込み済の場合にデータを読み出すおよび制御フラグをクリアし、プロセス間通信を行なうようにしている。

【0019】また、図5に示すように、送信プロセスを構成するデータ書き込み部25が専用バッファ19に送信しようとするデータを書き込んだ後、特別シグナルを受信プロセスに送信し、この特別シグナルを受信した受信プロセスを構成するデータ読み出し部32が専用バッファ19からデータを読み出し、プロセス間通信を行なうようにしている。

【0020】従って、送信プロセスから受信プロセスに通信が行われている中で発生した優先処理に必要なデータを優先的に通信することが可能となる。

【0021】

【実施例】次に、図1から図9を用いて本発明の実施例の構成および動作を順次詳細に説明する。

【0022】図1は、本発明の1実施例構成図を示す。図1において、送信プロセスは、データを受信プロセスに送信するものであって、ここでは、メッセージ送信部1から構成されるものである。

【0023】メッセージ送信部1は、メッセージをメッセージ受信部8に送信するものであって、ここでは、メッセージ領域獲得部3、データ書き込み部4、およびメッセージキュー登録部5などから構成されるものである。

【0024】メッセージ領域獲得部3は、メッセージバッファ2からデータを書き込む領域を獲得するものである。データ書き込み部4は、メッセージ領域獲得部3によってメッセージバッファ2から獲得した領域にデータを書き込むものである。

【0025】メッセージキュー登録部5は、メッセージバッファ2中の領域に書き込んだデータの情報（メッセージの優先度、メッセージ長、データのアドレスなど）を設定したメッセージを、メッセージキュー21に登録

5

するものである。

【0026】メッセージバッファ2は、メッセージ領域獲得部3が送信しようとするデータを格納する領域を獲得するためのバッファである。メッセージキュー21は、メッセージを登録する領域である。

【0027】カーネル6は、送信プロセスおよび受信プロセスなどを統括制御するものであって、ここでは、メッセージ送信部1がデータをメッセージバッファ2の領域に書き込むおよびメッセージをメッセージキュー21に登録した後、シグナルをシグナルハンドラー7を経由してメッセージ受信部8を含む受信プロセスに通知するための各種制御を行なうものである。

【0028】シグナルハンドラー7は、カーネル6からのシグナルを受信プロセスに通知するものである。シグナルが受信されたことは、メッセージ受信部に通知される。メッセージ受信部8は、メッセージ送信部1から送信されてきたシグナルを受信し、メッセージキュー21から最も優先度の高いメッセージを取り出し、このメッセージをもとにメッセージバッファ2からデータを読み出すものであって、メッセージ解析部9およびデータ読み出し部12などから構成されるものである。

【0029】メッセージ解析部9は、メッセージキュー21に登録されているメッセージの優先度およびデータ属性、データのアドレスなどを解析するものであって、メッセージ優先度解析部10およびメッセージデータ属性解析部11などから構成されるものである。

【0030】メッセージ優先度解析部10は、メッセージキュー21に登録されているメッセージのうちの優先度の最も高いメッセージを見つけたり、優先度の高いメッセージが複数あるあった場合には最も速く登録されたメッセージを見つけたりするものである。

【0031】メッセージデータ属性解析部11は、メッセージ優先度解析部10によって、見つけた最も優先度の高いメッセージのデータ属性、データのアドレスを解析するものである。

【0032】データ読み出し部12は、最も優先度の高いメッセージをもとに、対応するデータをメッセージバッファ2中の該当する領域から読み出すものである。次に、図2のフローチャートに示す順序に従い、図1の構成の動作を詳細に説明する。

【0033】図2において、S1は、バッファ（メッセージバッファ2）の中からデータ（メッセージデータ）を書き込むための領域を獲得する。S2は、データを書き込む。

【0034】S3は、データの優先度を高く設定する。これは、プロセス間で通信するデータの優先度を高くメッセージに設定する。S4は、メッセージキュー21に登録する。これは、S3で優先度を高く設定したメッセージを、メッセージキュー21に登録する。

【0035】S5は、受信プロセスにシグナルを送る。

6

このシグナルを受信すると、受信プロセスがS11からS17の処理を実行する。以上によって、送信側の送信プロセスがメッセージバッファ2からデータを書き込む領域を獲得してこれにデータを書き込むと共に、優先度を高く設定したメッセージをメッセージキュー21に登録した後、シグナルを受信プロセスに送信する。これにより、送信プロセスは優先度を高く設定してメッセージを受信プロセスに送信したこととなる。

【0036】S11は、メッセージが1つしかないか判別する。これは、S5で送られたシグナルを受信した受信プロセスが、メッセージキュー21を参照して自身の受信プロセス宛のメッセージが1つしかないか判別する。YESの場合には、S12でメッセージをメッセージキュー21から取り出し、S16でこのメッセージに従いメッセージバッファ2中のデータ領域からデータを読み出して処理し、S17でデータ領域を開放する。一方、S11でNOの場合には、S13に進む。

【0037】S13は、優先度の一番高いメッセージを探す。これは、S11のNOでメッセージキュー21に複数のメッセージが登録されていると判明したので、そのうちの優先度の一番高いメッセージを探す。

【0038】S14は、S13で探した結果、2つ以上あるか判別する。YESの場合には、優先度の一番高いメッセージが2つ以上あると判明したので、S15で早く来た方のメッセージをメッセージキュー21から取り出し、S16でこのメッセージに従いメッセージバッファ2中のデータ領域からデータを読み出して処理し、S17でデータ領域を開放する。一方、S14でNOの場合には、一番高い優先度のメッセージが1つと判明したので、S16でこのメッセージに従いメッセージバッファ2中のデータ領域からデータを読み出して処理し、S17でデータ領域を開放する。

【0039】以上によって、シグナルを受信した受信プロセスがメッセージキュー21を参照して1つしかメッセージがないときはそのメッセージを取り出し、2つ以上のメッセージがあつてかつ優先度の一番高いメッセージが2つ以上あつたときはそのうちの最も早く来たメッセージを取り出し、優先度の一番高いメッセージが1つのときにはこのメッセージを取り出し、優先度の高いメッセージがないときは最も早く来たメッセージを取り出し、これら取り出したメッセージをもとにメッセージバッファ2のデータ領域からデータを読み出し、当該データ領域を開放する。これにより、受信プロセス側で優先度が高く設定されたメッセージを優先的にメッセージキュー21から取り出してそのデータを読み出すことが可能となり、プロセス間で通信中に優先的にデータを送信プロセスから受信プロセスに渡すことが可能となる。

【0040】図3は、本発明の他の実施例構成図を示す。これは、専用バッファ19を設けてこれに優先度の高いデータを書き込んで受信プロセスに渡す構成であつ

て、通常（優先度が高くない）データの場合の書き込む領域の獲得、開放という処理が不要となり、処理の高速化を更に図ることができる。

【0041】図3において、メッセージ送信部14は、送信プロセス側にあり、データを専用バッファ19に書き込むデータ書き込み部15などから構成されるものである。

【0042】専用バッファ19は、データ書き込み部15が、受信プロセスに送信しようとするデータを書き込む、専用のバッファである。この専用バッファ19に

・制御フラグ

・データ（メッセージデータ）

を対にして書き込む。ここで、制御フラグは、データの書き込み済を表すものである。データ書き込み部15がデータを専用バッファ19に書き込んだときに当該書き込んだ旨を表すために、制御フラグを1にセットする。

【0043】メッセージ受信部16は、専用バッファ19からデータを読み出ししたりなどするものであって、データ読み込み部17およびタイマ18などから構成されるものである。

【0044】データ読み込み部17は、専用バッファ19の制御フラグが“1”か否かを判別し、“1”のときにデータを読み出した後、“0”にクリアするものである。制御フラグが“1”のデータを全て読み出した後、タイマ18で所定時間経過後に再度、制御フラグが

“1”か否かをチェックすることを繰り返す。

【0045】タイマ18は、所定時間毎にデータ読み込み部17に割り込みで通知し、専用バッファ13からデータの読み込みを実行させるものである。次に、図4の

フローチャートに示す順序に従い、図3の構成の動作を詳細に説明する。

【0046】図4の（a）は、フローチャートを示す。図4の（a）において、S21は、送信プロセスがデータを得る。S22は、データにフラグをつけ（制御フラグを“1”にセット）、バッファ（専用バッファ19）に書き込む。

【0047】S23は、次のデータのフラグをクリア（制御フラグを“0”にクリア）し、一連のデータの終わりの記号を付与する（EOFを付与する）。そして、S21に戻り、繰り返す。

【0048】以上によって、送信プロセスが送信依頼を受けたデータの制御フラグを“1”にして専用バッファ19に書き込み、受信プロセスに送信する。S31は、受信プロセスがデータが用意されているか判別する。これは、受信プロセスが専用バッファ19を参照してデータの制御フラグが“1”のものがあるか判別する。YESの場合には、S32で制御フラグが“1”のデータを読み込み、S33でデータを処理し、S34でその制御フラグを“0”にクリアし、S31に戻る。一方、S3

1のNOの場合には、S35で少し待ち、S36に進む。

【0049】S36は、一定時間以上経過したか判別する。YESの場合には、受信処理が終了したと判断し、終わる（END）。NOの場合には、S31に戻り、繰り返す。

【0050】以上によって、受信プロセスが専用バッファ19を参照して制御フラグが“1”のデータについて順次読み出して“0”クリアすることを繰り返し、送信されてきたデータを読み出す。これにより、送信プロセスから送られてきたデータを専用バッファ19を介して受信プロセスが迅速に受け取る、即ちメッセージバッファからデータ領域を獲得したり、開放したりする処理が不要となり、優先度の高いデータを当該専用バッファ19に書き込んで受信プロセスに渡すことにより、高速にプロセス間通信を行なうことが可能となる。

【0051】図4の（b）は、専用バッファの例を示す。ここでは、専用バッファ19には、制御フラグ+データを1対にして書き込むようにし、制御フラグが“1”のときにデータの書き込み済を表し、制御フラグが“0”のときはデータの読み出し済を表す。尚、一連のデータの最終の位置の制御フラグに“0”を書き込み、データの終わりである旨をセットしておく。

【0052】図5は、本発明の他の実施例構成図を示す。これは、従来のメッセージバッファ2および専用バッファ19を設け、優先度の高い特別データは専用バッファ19に書き込んだ後に特別シグナルを受信プロセスに送信し、当該専用バッファ19からデータを読み出し、一方、優先度の高くない通常のデータはメッセージバッファ2に書き込みおよびメッセージをメッセージキュー21に登録した後にシグナルを受信プロセスに送信し、メッセージキュー21からメッセージを取り出してこれをもとにメッセージバッファ2からデータを読み出すときの構成図である。

【0053】図5において、メッセージ送信部23は、送信プロセス側にあり、メッセージ領域獲得部24、データ書き込み部25、およびメッセージキュー登録部26などから構成されるものである。

【0054】メッセージ領域獲得部24は、メッセージバッファ2からデータを書き込むデータ領域を獲得するものである。データ書き込み部25は、メッセージバッファ2内の獲得したデータ領域にデータを書き込んだり、専用バッファ19にデータを書き込んだりするものである。

【0055】メッセージキュー登録部26は、メッセージをメッセージキュー21に登録して送信依頼したりするものである。専用バッファ19は、優先度の高い特別データを書き込むものである。

【0056】カーネル34は、送信プロセスおよび受信プロセスなどを統括制御するものであって、送信プロセ

スからのシグナルを受信プロセスに送るものである。シグナルハンドラ35は、送信プロセスから送られたシグナルを受信するためのものである。

【0057】メッセージ受信部28は、受信プロセス側にあり、メッセージ解析部29、データ読み出し部32、およびタイマ33などから構成されるものである。メッセージ解析部29は、メッセージキュー21に登録されているメッセージを解析するものであって、メッセージ優先度解析部30およびメッセージデータ属性解析部31から構成されるものである。

【0058】メッセージ優先度解析部30は、メッセージキュー21に登録されているメッセージの優先度を解析し、最も優先度の高いメッセージを取り出したり、最も優先度の高いメッセージが複数あったときは最も早く登録したメッセージを取り出したりなどするものである。

【0059】メッセージデータ属性解析部31は、メッセージデータの属性を解析するものである。データ読み出し部32は、メッセージバッファ2のデータ領域あるいは専用バッファ19からデータを読み出すものである。

【0060】タイマ33は、所定時間経過した旨を計測するものである。まず、図6を用いて図5の構成の専用バッファを用いた概念を説明する。

(1) 特別データの場合：

(1-1) 送信プロセスが特別データを専用バッファ19に書き込むおよび制御フラグを“1”にセットした後、特別シグナルを受信プロセスに送信する。

【0061】(1-2) 特別シグナルを受信した受信プロセスが、専用バッファ19から制御フラグが“1”のデータを順次読み出すと共に制御フラグを“0”クリアする。

【0062】以上によって、特別データについて、専用バッファ19を経由して受信プロセスに高速に送信することが可能となる。

(2) 通常のデータの場合：

(2-1) 送信プロセスが通常のデータを、メッセージバッファ2中の獲得したデータ領域に書き込むと共にメッセージをメッセージキュー21に登録した後、シグナルを受信プロセスに送信する。

【0063】(2-2) シグナルを受信した受信プロセスがメッセージキュー21から取り出したメッセージをもとにメッセージバッファ2中のデータ領域からデータを読み出す。

【0064】以上によって、通常のデータについて、メッセージバッファ2を経由して受信プロセスに送信することが可能となる。次に、図7のフローチャートに示す順序に従い、図5の構成の動作を詳細に説明する。

【0065】図7の(a)は、初期設定フローチャートを示す。図7の(a)において、S41は、専用のバッ

ファ(専用バッファ19)を獲得する。これは、図5の専用バッファ19を予め獲得し、送信の都度、獲得・開放を繰り返すときの処理を省略し、高速化を図るためである。

【0066】図7の(b)は、データの送受信フローチャートを示す。図7の(b)において、S51は、送信プロセスが特別なデータか判別する。YESの場合には、S52で専用のバッファ(専用バッファ19)に書き込み、S53で受信プロセスに特別なシグナルを送る。一方、NOの場合には、通常のデータであるので、S54でメッセージバッファ2からデータ領域を獲得してデータを書き込み、S55でメッセージをメッセージキュー21に登録し、S56で受信プロセスにシグナルを送る。

【0067】以上によって、特別データの場合に専用バッファ19に書き込んだ後に特別シグナルを受信プロセスに送り、一方、通常のデータの場合にメッセージバッファ2から獲得したデータ領域に書き込みおよびメッセージをメッセージキュー21に登録した後にシグナルを受信プロセスに送る。これらにより、特別データあるいは通常のデータを分けてそれぞれのデータの書き込みおよび受信プロセスに通知することが可能となる。

【0068】S61は、受信プロセスが特別なシグナルを受信したか判別する。YESの場合には、S62で専用バッファ19からデータを読み出し、S63でデータを処理する。一方、NOの場合には、通常のデータと判明したので、S64でメッセージをメッセージキュー21から取り出し、S65でこのメッセージをもとにメッセージバッファ2中のデータ領域からデータを読み出し、S66でデータを処理し、S67でデータ領域を開放する。

【0069】以上によって、受信したシグナルが特別シグナルの場合に専用バッファ19からデータを読み出して優先度の高いデータを高速にプロセス間通信できる。一方、通常のシグナルの場合にメッセージバッファ2のデータ領域からデータを読み出し、通常のプロセス間通信できる。これらにより、優先的にプロセス間通信する必要のあるデータは、特別データとして専用バッファ19経由で高速にデータ通信を行い、一方、通常のデータはメッセージバッファ2を経由してデータ通信することが可能となる。

【0070】図8は、本発明の他の動作説明図を示す。これは、特別シグナルの代わりに、通信すべき第1番目のデータを優先度を高くして送り、第2番目以降のデータを専用バッファを使って送信する他の動作説明である。

【0071】図8の(a)は、初期設定のフローチャートを示す。図8の(a)において、専用のバッファ(専用バッファ19)を獲得する。これにより、図5の専用バッファ19が獲得できたこととなる。

11

【0072】図8の(b)は、送信プロセスの動作フローチャートを示す。図8の(b)において、S81は、特別なデータか判別する。YESの場合には、S82で第1番目か判別し、YESのときにS83に進み、NOのときにS87でフラグ(制御フラグ)を1にし、フラグ共々専用バッファに書き込み、終了する(END)。一方、NOの場合には、S83に進む。

【0073】S83は、領域を獲得し、データを書き込む(メッセージバッファ2中の獲得したデータ領域にデータを書き込む)。S84は、メッセージの優先度を高く設定する。

【0074】S85は、S84で優先度を高く設定したメッセージをメッセージキュー21に登録する。S86は、受信プロセスにシグナルを送る。

【0075】以上によって、特別データかつ第1番目の場合にメッセージバッファ2から獲得したデータ領域にデータを書き込むおよび優先度を高く設定したメッセージをメッセージキュー21に登録し、一方、特別データかつ第2番目以降の場合にデータを専用バッファ19に書き込む。これにより、特別データの第1番目はメッセージバッファ2に書き込んで受信プロセスに送信し、特別データの第2番目以降は専用バッファ19に書き込んで受信プロセスに送信する。尚、特別データでない通常のデータは、常にメッセージバッファ2に書き込んで受信プロセスに送信する。

【0076】図8の(c)は、受信プロセスの動作フローチャートを示す。図8の(c)において、S91は、S86で送られたシグナルを受信した受信プロセスがメッセージキュー21からメッセージを参照して優先度、順番に従いメッセージを取り出す。これは、シグナルを受信した受信プロセスがメッセージキュー21を参照して優先度の高いメッセージが1つの場合にはこのメッセージを取り出し、複数の優先度の高いメッセージがある場合には最も早く登録されたメッセージを取り出し、あるいは優先度の高いメッセージがない場合には早く登録したメッセージを取り出す。

【0077】S92は、データを読み出す。これは、S91で取り出したメッセージをもとにメッセージバッファ2中から該当するデータ領域からデータを読み出す。S93は、特別なデータか判別する。YESの場合には、S94でデータを処理し、S95に進む。一方、NOの場合には、通常のデータであったのでS99でデータを処理する。

【0078】S95は、専用バッファにデータが用意されているか判別する。これは、S87で制御フラグが“1”にセットされたデータが専用バッファ19に書き込まれているか判別する。YESの場合には、S96で専用バッファ19から制御フラグ“1”のデータを読み込み、S97でデータを処理し、S98で制御フラグを“0”にクリアし、S95に戻る。一方、NOの場合に

12

は、S100で少しまち、S101で一定時間以上経過したか判別し、YESのときに一連のデータ受信処理を終了し、NOのときにS95に戻る。

【0079】以上によって、受信プロセッサが優先度の高いメッセージをメッセージキュー21から優先的に取り出し、この取り出したメッセージをもとにメッセージバッファ2からデータを読み出し、特別なデータの場合に専用バッファ19からこれに続く第2番目以降のデータを順次終わりまで読み出す。これらにより、特別なデータの第1番目をメッセージバッファ2を介して受信プロセスに送り、特別データの第2番目以降を専用バッファ19を介して受信プロセスに送ることが可能となる。これらにより、特別なシグナルを送ることなく、専用バッファを使用して高速にプロセス間通信を行なうことが可能となる。

【0080】図9は、本発明のペンの座標データ例を示す。図9の(a)は、専用バッファ19の例を示す。この専用バッファ19には、図示の下記のように、

・制御フラグ(1:書き込み済のデータ有り、0:読み出し済/終了

・X座標、Y座標(座標データ)

・ペンの状態(属性)

を1レコードとして書き込む。これらのデータを書き込んだときに制御フラグを“1”にセットする。

【0081】図9の(b)は、1座標点データの例を示す。この1座標点データは、図示の下記のようである。この例は、ペン入力に適用した例である。電子ペンがタブレットの感応領域に入ると、50~200ポイント/秒の割合で、1座標点毎にCPUに割り込みが発生するものとする(即ち、1秒間に50~200回の割り込みが発生するものとする)。ここで、ペン座標を読み取るプロセスを送信プロセスとし、この読み取った座標を受信プロセスに送信する。

【0082】(1) ユーザがペンをタブレットの感応領域に入れると、割り込みが発生し、送信プロセスが第1点目の1座標分のペン情報2を得る。座標データを得た送信プロセスは、メッセージバッファ2からデータ領域を獲得して座標データを書き込むおよび優先度を高く設定したメッセージをメッセージキュー21に登録した後、シグナルを受信プロセスに送る。次の第2点目からユーザがペンをタブレットの感応領域から離すまでの間に得られた座標データを制御フラグ“1”と一緒に専用バッファ19に順次書き込む。この際、1点分の制御フラグ“1”と座標データを書き込むと、次の制御フラグに“0”クリアして終わりの記号を書き込む。

【0083】(2) (1)からのシグナルを受信した受信プロセスは、メッセージキュー21にたまっているメッセージの中から一番優先度の高いメッセージを取り出す。このメッセージをもとにメッセージバッファ2の該当データ領域からデータ(座標データ)を読み出し、

13

座標データの処理を行った後、専用バッファ19から制御フラグ“1”の第2点目以降のデータ（座標データ）を順次読み出す。この際、制御フラグが“1”でなくて未だ書き込まれていない場合には、少し待つ（例えばペンのレートが50ポイント/秒とすると、約0.02秒で1座標分のデータが得られることとなるので、0.02秒待つ）。再び専用バッファ19からデータを読み出す。連続して一定時間（例えば0.5秒以上）待っても制御フラグ“1”とならない場合には、ペンがタブレットの感応領域から離れたものと判断し、終了する。

【0084】以上によって、ペンを用いてタブレットから座標データを入力する場合、最初の第1点目の座標データはメッセージバッファ2経由して優先度を高くして受信プロセスに送信し、第2点間から専用バッファ19を経由して高速に座標データを受信プロセスに送信する。これらにより、特別なシグナルなしに高速に座標データを送信プロセスから受信プロセスに送信することが可能となる。

【0085】

【発明の効果】以上説明したように、本発明によれば、データをメッセージバッファ2のデータ領域に書き込むと共にメッセージに優先度を高く設定してメッセージキュー21に登録した後にシグナルを受信プロセスに送り、シグナルを受け取った受信プロセスがメッセージキュー21から優先度の高いメッセージを取り出し、これをもとにメッセージバッファ2中のデータ領域からデータを読み出したり、専用バッファを設けてこれにデータを書き込んで特別シグナルを受信プロセスに送り、特別シグナルを受け取って受信プロセスが専用バッファからデータを読み出したり、両者を組み合わせたりし、優先的にデータを送信プロセスから受信プロセスに送信する構成を採用しているため、送信プロセスから受信プロセスに通信が行われている中で発生した優先処理に必要な

14

データを優先的に渡すことができる。特に、メッセージ（データ）に優先度を設けたり、専用バッファを設けたり、両者を併用することにより、優先的に処理したい通信を高速に処理できると共に、非常に高い頻度で発生するデータの通信を処理することができる。

【図面の簡単な説明】

【図1】本発明の1実施例構成図である。

【図2】本発明の動作説明フローチャートである。

【図3】本発明の他の実施例構成図である。

【図4】本発明の他の実施例動作説明図である。

【図5】本発明の他の実施例構成図である。

【図6】本発明の専用バッファを用いた概念説明図である。

【図7】本発明の他の動作説明図である。

【図8】本発明の他の動作説明図である。

【図9】本発明のペンの座標データ例である。

【図10】従来技術の説明図である。

【符号の説明】

1、14、23：メッセージ送信部

2：メッセージバッファ

21：メッセージキュー

3、24：メッセージ領域獲得部

4、15、25：データ書き込み部

5、26：メッセージキュー登録部

6：カーネル

8、16、28：メッセージ受信部

9、29：メッセージ解析部

10、30：メッセージ優先度解析部

11、31：メッセージデータ属性解析部

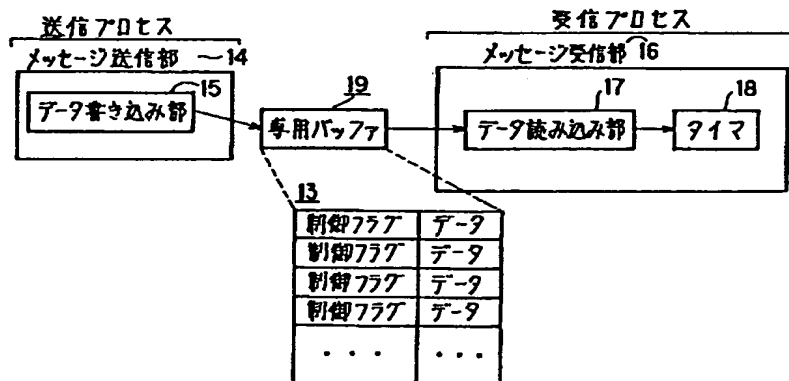
12、17、32：データ読み出し部

19：専用バッファ

18、33：タイマ

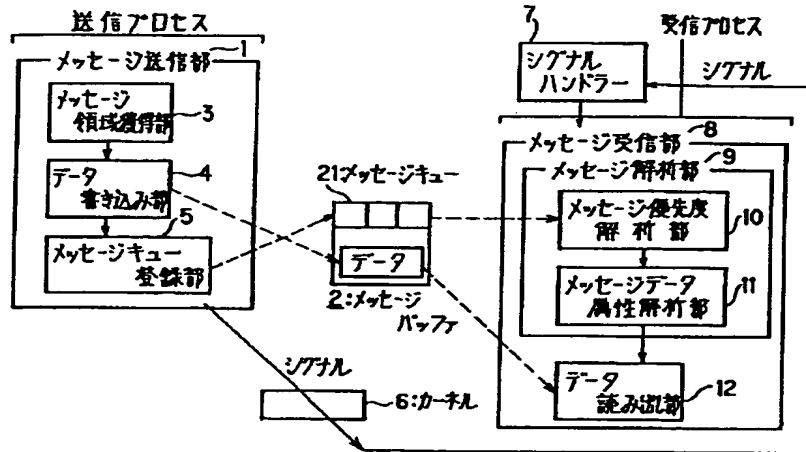
【図3】

本発明の他の実施例構成図



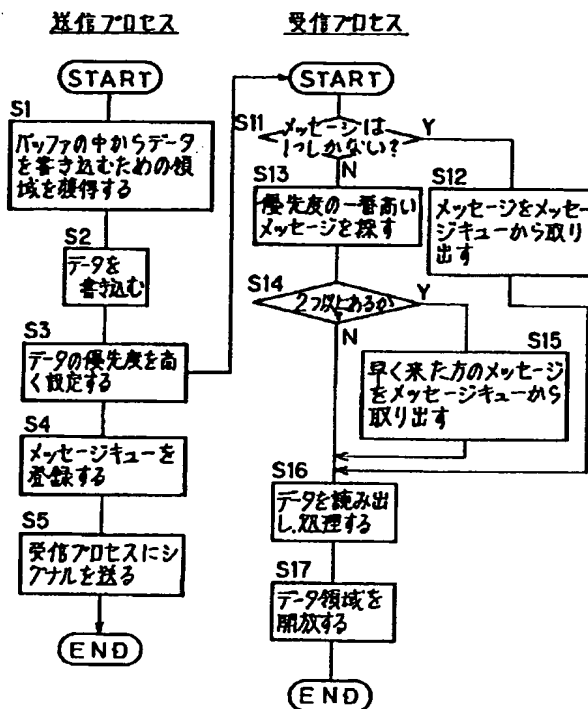
【図1】

本発明の1実施例構成図



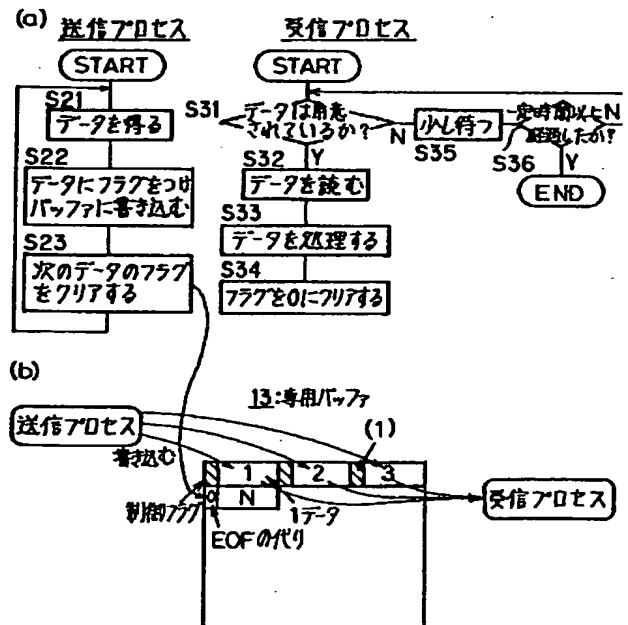
【図2】

本発明の動作説明フローチャート



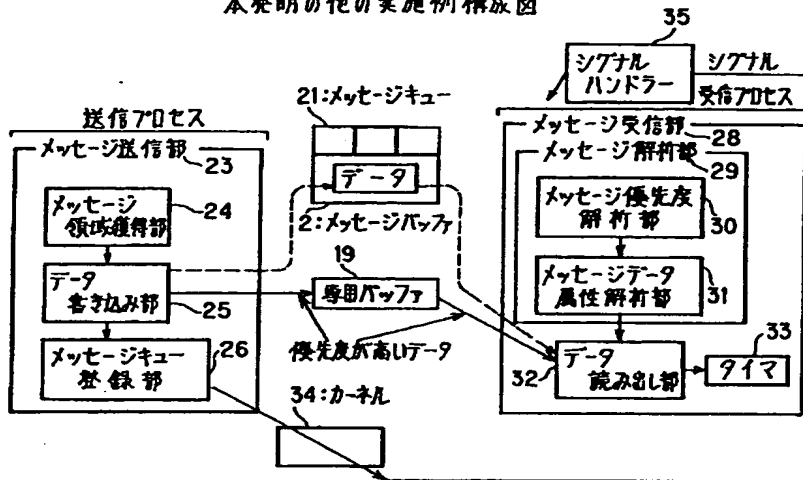
【図4】

本発明の他の実施例動作説明図



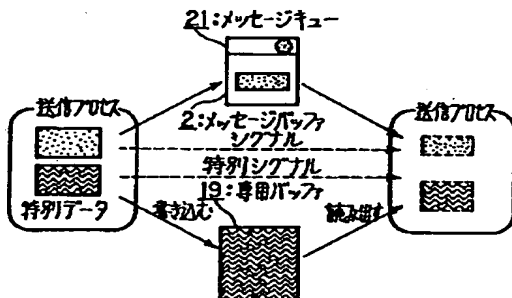
【図5】

本発明の他の実施例構成図



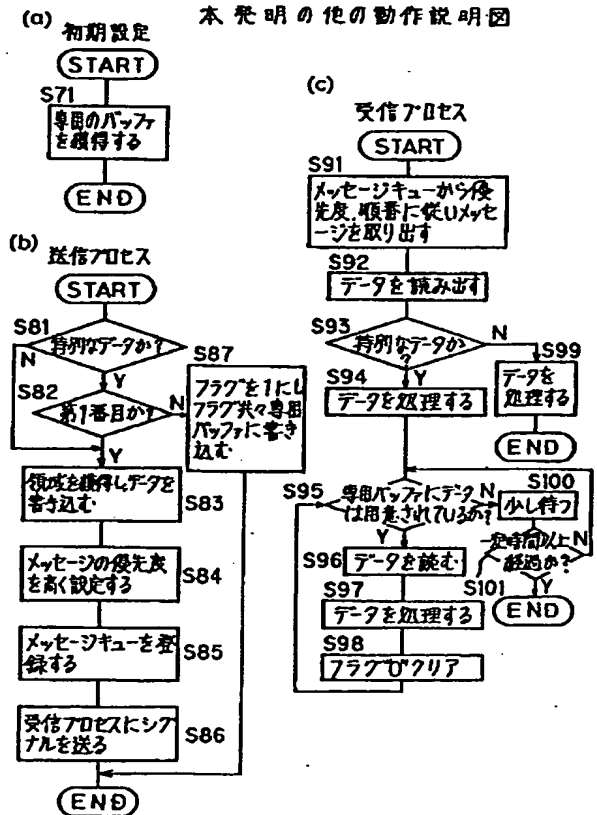
【図6】

本発明の専用バッファを用いた概念説明図

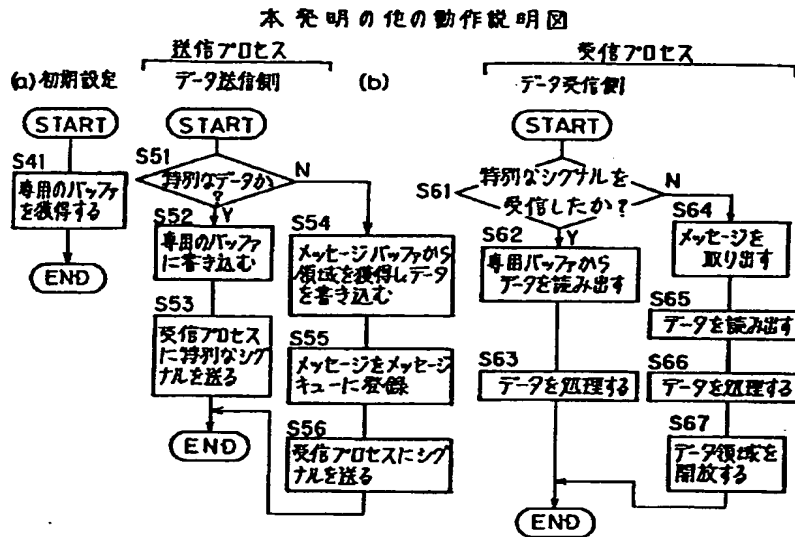


【図8】

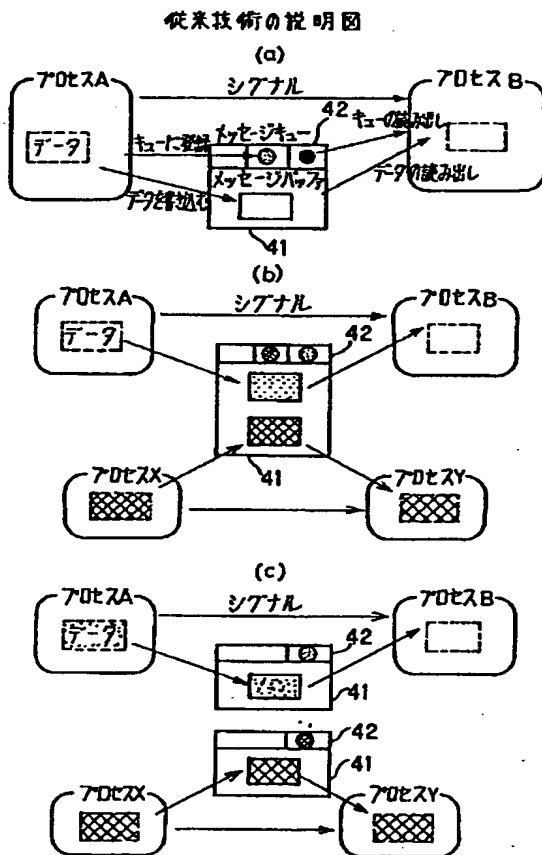
本発明の他の動作説明図



【図7】



【図10】



【図9】

本発明のペンの座標データ例

(a)

19:専用バッファ

制御フラグ	1	X座標	Y座標	ペンの状態
	0			

(b)

1座標データ

制御フラグ	X座標	Y座標	ペンの状態
-------	-----	-----	-------

JP 07-084804

[COMPUTER TRANSLATION FROM JAPANESE PATENT OFFICE WEB SITE]

[Detailed Description of the Invention]

[0001]

[Industrial Application] This invention is an interprocess communication method which communicates between processes, and relates to the interprocess communication method using the interprocess communication function on the multi-process OS in a calculating machine.

[0002]

[Description of the Prior Art] Interprocess communication is prepared as an approach of exchanging data between the multiple processes performed in parallel under the environment of Multiprocessor OS. There are a signal, a message, a shared memory, a semaphore, etc. as an interprocess communication function. In order to transmit and receive much data of unspecified size freely between multiple processes, data are usually exchanged using the message shown in (a) of drawing 10 . If it explains briefly (1) While Process A writes data in the field gained from the message buffer 41, after registering a message into a message queue 42, a signal is sent to the process B of a reception place.

[0003] (2) The process B which received the signal takes out a message from a message queue 42, and reads data from a message buffer 41 based on this. It means that data were passed to Process B from Process A by this.

[0004] (b) of drawing 10 shows the example which prepared one accessible message buffer from all processes. Actuation is explained.

(1) Prepare one accessible message buffer 41 from all the processes related to application. When communicating, the empty field for required size is gained from a message buffer 41, and while the process of a transmitting side writes data in this empty field, after registering a message into a message queue 42, a signal is transmitted to the process of a receiving side.

[0005] (2) The process of the receiving side which received the signal takes out a message from a message queue 42, and reads data from a message buffer 41 based on this.

[0006] (c) of drawing 10 shows the example for which the process and the process prepared the message buffer 41 corresponding to 1 to 1. Actuation is explained.

(1) All the processes related to application prepare a process and the message buffer 41 corresponding to 1 to 1, respectively. After registering a message into a message queue 42 while the process of a transmitting side gains the empty field of the message buffer 41 of the process of a receiving side and writes data in this when communicating, a signal is transmitted to the process of a receiving side.

[0007] (2) The process of the receiving side which received the signal sends a message from a message queue 42 picking one by one, and reads data from a message buffer 41 in order of this.

[0008]

[Problem(s) to be Solved by the Invention] According to the conventional interprocess communication mentioned above, there was a problem that communications processing could not be preferentially performed even if a communication link to process

preferentially occurs. For example, when drawing on a display by the pen computer using an electronic pencil, the coordinate data of a pen must be processed by top priority. If delay occurs in a communication link, the situation which draws several mm back of a pen happens. Now, it becomes a big problem on the human interface of a pen that it is hard to use.

[0009] This invention aims at communicating preferentially the required data of a priority processing generated while the communication link was carried out to other processes from the process in order to solve these problems.

[0010]

[Means for Solving the Problem] With reference to drawing 1 , drawing 3 , and drawing 5 , The means for solving a technical problem is explained. In drawing 1 , drawing 3 , and drawing 5 , a message buffer 2 writes in data. A message queue 21 registers the message to which a setup etc. carried out priority highly.

[0011] The message region acquisition section 3 gains the field which stores data from a message buffer 2. The data write-in section 4 writes data in the field gained out of the message buffer 2.

[0012] The message queue registration section 5 registers into a message queue 21 the message to which a setup etc. carried out priority highly. The message priority analysis section 10 finds a message with a high priority (or message most registered early when there was a message with two or more high priorities) out of the message registered into the message queue 21.

[0013] The data read-out section 12 reads data from the field of a message buffer 2. The data write-in section 15 writes writing and a control flag in the exclusive buffer 19, and sets to it the data which it is going to transmit settled.

[0014] The control flag in the exclusive buffer 19 writes in the data read-out section 17, and it distinguishes whether it is settled for every predetermined time, and when finishing [writing], it is a thing which reads data and which reaches and clears a control flag.

[0015] Data, the control flag with which it writes in and expresses settled are written in the exclusive buffer 19. The data write-in section 25 writes in the data which it is going to transmit to the exclusive buffer 19.

[0016] The data read-out section 32 reads data from the exclusive buffer 19.

[0017]

[Function] As shown in drawing 1 , as for this invention, the message region acquisition section 3 which constitutes a transmitting process gains a field from a message buffer 2. The data which it is going to transmit to the field which the data write-in section 4 gained are written in. After the message queue registration section 5 registers into a message queue 21 the message which set up the address of the priority about data, and data etc., A signal is transmitted to a receiving process. A message with a high priority (or message most registered early when there was a message with two or more high priorities) is found out of the message by which the message priority analysis section 10 which constitutes the receiving process which received this signal is registered into the message queue 21. The message data attribute analysis section 11 analyzes a message, reads the address of data from the field of delivery and a message buffer 2 to the data read-out section 12, and is made to perform interprocess communication.

[0018] Moreover, writing and a control flag are written in and the data which the data write-in section 15 which constitutes a transmitting process as shown in drawing 3 tends to

transmit to the exclusive buffer 19 set to settled, and when finishing [writing], and the control flag in the exclusive buffer 19 writes in, and the data read-out section 17 which constitutes a receiving process distinguishes whether it is settled for every predetermined time, and it reads data, a control flag is cleared and it is made to carry out interprocess communication.

[0019] Moreover, as shown in drawing 5 , after the data write-in section 25 which constitutes a transmitting process writes in the data which it is going to transmit to the exclusive buffer 19, a signal is specially transmitted to a receiving process, and the data read-out section 32 which constitutes the receiving process which received this special signal reads data from the exclusive buffer 19, and is made to perform interprocess communication.

[0020] Therefore, it becomes possible to communicate preferentially data required for the priority processing generated while the communication link was carried out to the receiving process from the transmitting process.

[0021]

[Example] Next, the configuration and actuation of the example of this invention are explained to a detail one by one using drawing 9 from drawing 1 .

[0022] Drawing 1 shows 1 example block diagram of this invention. In drawing 1 , a transmitting process transmits data to a receiving process, and consists of the message-sending sections 1 here.

[0023] The message-sending section 1 transmits a message to the message receive section 8, and consists of the message region acquisition section 3, the data write-in section 4, the message queue registration section 5, etc. here.

[0024] The message region acquisition section 3 gains the field which writes in data from a message buffer 2. The data write-in section 4 writes data in the field gained from the message buffer 2 by the message region acquisition section 3.

[0025] The message queue registration section 5 registers into a message queue 21 the message which set up the information on the data written in the field in a message buffer 2 (the priority of a message, message length, address of data, etc.).

[0026] A message buffer 2 is a buffer for gaining the field which stores the data which the message region acquisition section 3 tends to transmit. A message queue 21 is a field which registers a message.

[0027] A kernel 6 carries out generalization control of a transmitting process, the receiving process, etc., and after [when the message-sending section 1 writes data in the field of a message buffer 2] reaching and registering a message into a message queue 21, it performs various control for notifying a signal to the receiving process which includes the message receive section 8 via a signal handler 7 here.

[0028] A signal handler 7 notifies the signal from a kernel 6 to a receiving process. A message receive section is notified of the signal having been received. The message receive section 8 receives the signal transmitted from the message-sending section 1, takes out a message with the highest priority from a message queue 21, reads data from a message buffer 2 based on this message, and consists of the message analysis section 9, the data read-out section 12, etc.

[0029] The message analysis section 9 analyzes the address of the priority of the message registered into the message queue 21 and a data attribute, and data etc., and consists of the message priority analysis section 10, the message data attribute analysis section 11, etc.

[0030] The message priority analysis section 10 finds the highest message of the priority of the messages registered into the message queue 21, or when [that there are two or more messages with a high priority] it is, it finds the message registered most quickly.

[0031] The message data attribute analysis section 11 analyzes the data attribute of the found message with the highest priority, and the address of data by the message priority analysis section 10.

[0032] The data read-out section 12 reads data [/ based on a message with the highest priority] from the field where it corresponds in a message buffer 2. Next, actuation of the configuration of drawing 1 is explained to a detail according to the sequence shown in the flow chart of drawing 2 .

[0033] In drawing 2 , S1 gains the field for writing in data (message data) out of a buffer (message buffer 2). S2 writes in data.

[0034] S3 sets up the priority of data highly. This sets highly the priority of the data which communicate between processes as a message. S4 is registered into a message queue 21. This registers into a message queue 21 the message which set up the priority highly by S3.

[0035] S5 sends a signal to a receiving process. If this signal is received, a receiving process will perform processing of S11 to S17. While the transmitting process of a transmitting side writes in data by the above to gain the field which writes in data from a message buffer 2, after registering into a message queue 21 the message which set up the priority highly, a signal is transmitted to a receiving process. It means that a transmitting process sets up a priority highly and had transmitted the message to the receiving process by this.

[0036] S11 distinguishes whether there is only one message. The receiving process which received the signal with which this was sent by S5 distinguishes [the message addressed to an own receiving process] whether there is only one with reference to a message queue 21. In YES, a message is taken out from a message queue 21 by S12, data are read and processed from the data area in a message buffer 2 in it according to this message by S16, and a data area is opened by S17 in it. On the other hand, in NO, it progresses by S11 S13.

[0037] S13 looks for the highest message of a priority. Since this turned out to register two or more messages into the message queue 21 by NO of S11, it looks for the highest message of the priority of them.

[0038] S14 distinguishes whether there are two or more, as a result of searching by S13. Since it became clear that there are two or more highest messages of a priority in YES, the message of the direction which came by S15 early is taken out from a message queue 21, data are read and processed from the data area in a message buffer 2 according to this message by S16, and a data area is opened by S17. On the other hand, since the message of the highest priority was proved by S14 that it is one in NO, according to this message, data are read and processed from the data area in a message buffer 2 by S16, and a data area is opened by S17.

[0039] By the above, the receiving process which received the signal takes out the message, when there is only one message with reference to a message queue 21. The message by which it came of them early most when there were two or more messages and there were two or more highest messages of a priority is taken out. This message is taken out when the number of the highest messages of a priority is one. When there is no message with a high priority, the message which came early most is taken out, data are

read from the data area of a message buffer 2 based on these messages sent picking, and the data area concerned is opened. It becomes possible for a priority to take out by this the message set up highly from a message queue 21 preferentially by the receiving process side, and to read the data, and it becomes possible to pass data between processes to a receiving process from a transmitting process preferentially during a communication link.

[0040] Drawing 3 shows other example block diagrams of this invention. This is a configuration which writes in data with a high priority for the ability forming the exclusive buffer 19, and is passed to a receiving process, and processing called acquisition of the field written in in the case of being usually (a priority not being high) data and disconnection becomes unnecessary, and it can attain improvement in the speed of processing further.

[0041] In drawing 3, the message-sending section 14 is in a transmitting process side, and it consists of the data write-in sections 15 which write data in the exclusive buffer 19.

[0042] The exclusive buffer 19 is a buffer of dedication with which the data write-in section 15 writes in the data which it is going to transmit to a receiving process. As shown in this exclusive buffer 19 at the lower berth, it is - control flag data (message data).

It writes in by making it ****. Here, a control flag expresses the write-in settled one of data. Since the written-in purport concerned is expressed when the data write-in section 15 writes data in the exclusive buffer 19, a control flag is set to 1.

[0043] that the message receive section 16 reads data from the exclusive buffer 19 **** – etc. – it carries out and consists of the data reading section 17, a timer 18, etc.

[0044] The data reading section 17 is cleared to "0", after the control flag of the exclusive buffer 19 distinguishes whether it is "1" and reads data at the time of "1." After a control flag reads all the data of "1", it repeats again that a control flag confirms whether to be "1" or not after predetermined time progress with a timer 18.

[0045] A timer 18 is notified to the data reading section 17 by interruption for every predetermined time, and performs reading of data from the exclusive buffer 13. Next, actuation of the configuration of drawing 3 is explained to a detail according to the sequence shown in the flow chart of drawing 4.

[0046] (a) of drawing 4 shows a flow chart. In (a) of drawing 4, as for S21, a transmitting process obtains data. S22 attaches a flag to data (a control flag is set to "1"), and writes it in a buffer (exclusive buffer 19).

[0047] S23 clears the flag of the following data (clear [to "0"] in a control flag), and gives the notation of an series of ends of data (EOF is given). And it returns and repeats to S21.

[0048] By the above, a transmitting process sets to "1" the control flag of the data which received the transmitting request, and it writes in the exclusive buffer 19, and transmits to a receiving process. As for S31, a receiving process distinguishes whether data are prepared. As for this, a receiving process distinguishes whether the thing of "1" has the control flag of data with reference to the exclusive buffer 19. In YES, a control flag reads the data of "1" by S32, data are processed by S33, the control flag is cleared to "0" by S34, and it returns to S31. On the other hand, in NO of S31, it progresses for a while by S35 waiting and S36.

[0049] S36 distinguishes whether it passed beyond fixed time amount. In YES, it judges that reception was completed and finishes (END). In NO, it returns and repeats S31.

[0050] the above – a receiving process – the exclusive buffer 19 – referring to – a control flag – the data of "1" – one by one – reading appearance – carrying out – "0" – the data

which repeated clearing and have been transmitted are read. It enables this to perform interprocess communication at a high speed by the processing which a receiving process receives quickly the data sent from the transmitting process through the exclusive buffer 19, namely, gains or opens a data area wide from a message buffer becoming unnecessary, writing data with a high priority in the exclusive buffer 19 concerned, and passing a receiving process.

[0051] (b) of drawing 4 shows the example of an exclusive buffer. Here, control flag + data are made into one pair, and are written in, when a control flag is "1", the write-in settled one of data is expressed, and when a control flag is "0", it expresses the read-out settled of data with the exclusive buffer 19. In addition, "0" is written in the control flag of the last location of a series of data, and the purport which is the end of data is set.

[0052] Drawing 5 shows other example block diagrams of this invention. This forms a conventional message buffer 2 and the conventional exclusive buffer 19. After writing the special data with a high priority in the exclusive buffer 19, they transmit a signal to a receiving process specially. Read data from the exclusive buffer 19 concerned, and on the other hand, the usual data which are not expensive as for a priority transmit a signal to a receiving process, after registering writing and a message into a message buffer 2 at a message queue 21. It is a block diagram when taking out a message from a message queue 21 and reading data from a message buffer 2 based on this.

[0053] In drawing 5, the message-sending section 23 is in a transmitting process side, and consists of the message region acquisition section 24, the data write-in section 25, the message queue registration section 26, etc.

[0054] The message region acquisition section 24 gains the data area which writes in data from a message buffer 2. The data write-in section 25 writes data in the data area gained in the message buffer 2, or writes data in the exclusive buffer 19.

[0055] The message queue registration section 26 registers a message into a message queue 21, and carries out a transmitting request. The exclusive buffer 19 writes in special data with a high priority.

[0056] A kernel 34 carries out generalization control of a transmitting process, the receiving process, etc., and sends the signal from a transmitting process to a receiving process. A signal handler 35 is for receiving the signal sent from the transmitting process.

[0057] The message receive section 28 is in a receiving process side, and consists of the message analysis section 29, the data read-out section 32, a timer 33, etc. The message analysis section 29 analyzes the message registered into the message queue 21, and consists of the message priority analysis section 30 and the message data attribute analysis section 31.

[0058] taking out the message most registered early when the message priority analysis section 30 analyzed the priority of the message registered into the message queue 21, a message with the highest priority was taken out or there were two or more messages with the highest priority **** – etc. – it carries out.

[0059] The message data attribute analysis section 31 analyzes the attribute of message data. The data read-out section 32 reads data from the data area or the exclusive buffer 19 of a message buffer 2.

[0060] A timer 33 measures the purport which carried out predetermined time progress. First, the concept using the exclusive buffer of the configuration of drawing 5 is explained using drawing 6.

(1) In the case of data, it is : (1-1) specially. After [when a transmitting process writes data in the exclusive buffer 19 specially] reaching and setting a control flag to "1", a signal is specially transmitted to a receiving process.

[0061] (1-2) while a control flag reads [the receiving process which received the signal specially] the data of "1" from the exclusive buffer 19 one by one – a control flag – "0" – clear.

[0062] The above enables it to transmit to a high speed via the exclusive buffer 19 about data specially at a receiving process.

(2) In the case of the usual data, it is : (2-1). While a transmitting process writes the usual data in the data area gained in the message buffer 2, after registering a message into a message queue 21, a signal is transmitted to a receiving process.

[0063] (2-2) The receiving process which received the signal reads data from the data area in a message buffer 2 based on the message taken out from the message queue 21.

[0064] It becomes possible to transmit to a receiving process via a message buffer 2 about the usual data by the above. Next, actuation of the configuration of drawing 5 is explained to a detail according to the sequence shown in the flow chart of drawing 7.

[0065] (a) of drawing 7 shows an initialization flow chart. In (a) of drawing 7, S41 gains the buffer (exclusive buffer 19) of dedication. This is for gaining the exclusive buffer 19 of drawing 5 beforehand, omitting the processing when repeating acquisition and disconnection at every transmission, and attaining improvement in the speed.

[0066] (b) of drawing 7 shows the transceiver flow chart of data. In (b) of drawing 7, a transmitting process distinguishes S51 in special data. In YES, it writes in the buffer (exclusive buffer 19) of dedication by S52, and a signal special to a receiving process is sent by S53. On the other hand, since it is data usual to the case of NO, a data area is gained from a message buffer 2 by S54, data are written in, a message is registered into a message queue 21 by S55, and a signal is sent to a receiving process by S56.

[0067] After registering writing and a message into the data area which gained the signal from the message buffer 2 in the receiving process specially by the above delivery and in the case of the usual data on the other hand after writing in specially at the exclusive buffer 19 in the case of data at a message queue 21, a signal is sent to a receiving process. It becomes possible to divide data or the usual data specially and to notify to the writing and receiving process of each data by these.

[0068] S61 distinguishes whether the signal with a special receiving process was received. In YES, data are read from the exclusive buffer 19 by S62, and data are processed by S63 in it. On the other hand, since it was proved that it is the usual data, in NO, a message is taken out from a message queue 21 by S64, data are read from the data area in a message buffer 2 based on this message by S65 to it, data are processed by S66 to it, and a data area is opened by S67 to it.

[0069] By the above, when the received signal is a signal specially, data are read from the exclusive buffer 19 and the interprocess communication of the data with a high priority can be carried out to a high speed. the case of the usual signal on the other hand – the data area of a message buffer 2 to data – reading – the usual interprocess communication – it can do. By these, the data with the need of carrying out interprocess communication preferentially perform data communication at a high speed by exclusive buffer 19 course as data specially, and, on the other hand, the usual data become possible [carrying out data communication via a message buffer 2].

[0070] Drawing 8 shows other explanatory views of this invention of operation. This is other explanation of operation which makes a priority high for the 1st data which should communicate instead of a signal specially, and transmits the data of delivery and the 2nd henceforth using an exclusive buffer.

[0071] (a) of drawing 8 shows the flow chart of initial setting. The buffer (exclusive buffer 19) of dedication is gained in (a) of drawing 8 . By this, the exclusive buffer 19 of drawing 5 is able to be gained.

[0072] (b) of drawing 8 shows the operation flow chart of a transmitting process. In (b) of drawing 8 , S81 is distinguished in special data. the case of YES – S82 – the 1st – distinguishing – the time of YES – S83 – progressing – the time of NO – S87 – a flag (control flag) – 1 – carrying out – Flagg – it writes in an exclusive buffer together and ends (END). On the other hand, in NO, it progresses S83.

[0073] S83 gains a field and writes in data (data are written in the data area gained in the message buffer 2). S84 sets up the priority of a message highly.

[0074] S85 registers into a message queue 21 the message which set up the priority highly by S84. S86 sends a signal to a receiving process.

[0075] By the above, the message which writes data in the data area gained from the message buffer 2 to data and the 1st case and which reached and set up the priority highly is registered into a message queue 21, and, on the other hand, in data and 2nd henceforth, data are written in specially at the exclusive buffer 19. Thereby, the 1st of data is written in a message buffer 2, and it transmits to a receiving process, and specially, it writes in the exclusive buffer 19 2nd after data, and it is transmitted to a receiving process. In addition, the usual data which are not data specially are always written in a message buffer 2, and are transmitted to a receiving process.

[0076] (c) of drawing 8 shows the operation flow chart of a receiving process. In (c) of drawing 8 , the receiving process which received the signal with which S91 was sent by S86 takes out a message from a message queue 21 according to a priority and sequence with reference to a message. When this takes out this message when the number of messages with a high priority is one with reference to a message queue 21 for the receiving process which received the signal, the message registered most early is taken out when there is a message with two or more high priorities, or there is no message with a high priority, it takes out the message registered early.

[0077] S92 reads data. This reads data from the data area which corresponds out of a message buffer 2 based on the message taken out by S91. S93 is distinguished in special data. In YES, data are processed by S94, and it progresses to S95. On the other hand, in NO, since it was the usual data, data are processed by S99.

[0078] S95 distinguishes whether data are prepared for the exclusive buffer. This distinguishes whether the data with which control Flagg was set to "1" by S87 are written in the exclusive buffer 19. In YES, the data of control Flagg"1" are read from the exclusive buffer 19 by S96, data are processed by S97, control Flagg is cleared to "0" by S98, and it returns to S95. On the other hand, it waits for a while by S100, and in NO, it distinguishes whether it passed beyond fixed time amount in S101, a series of data receptions are ended to it at the time of YES, and it returns to S95 at the time of NO.

[0079] By the above, a receiving processor takes out a message with a high priority from a message queue 21 preferentially, data are read from a message buffer 2 based on this taken-out message, and the data of the 2nd henceforth which continues from the exclusive

buffer 19 at this in the case of special data are read to an end one by one. These enable it delivery and to send the 2nd data or subsequent ones to a receiving process through the exclusive buffer 19 specially through a message buffer 2 at a receiving process in the 1st of special data. It becomes possible to perform interprocess communication at a high speed by these using an exclusive buffer, without sending a special signal.

[0080] Drawing 9 shows the example of coordinate data of the pen of this invention. (a) of drawing 9 shows the example of the exclusive buffer 19. In this exclusive buffer 19, illustration is - control Flagg (1: those [finishing / writing] with data, read [0:]/termination and X coordinate, Y coordinate (coordinate data)) as follows.

- The condition of a pen (attribute)

It writes in as one record. Control Flagg is set to "1" when these data are written in.

[0081] (b) of drawing 9 shows the example of 1 coordinate point data. this 1 coordinate point data – illustration – being as follows . This example is an example applied to the pen input, and is **. If an electronic pencil goes into the influence field of a tablet, an interrupt shall occur in CPU for every 1 coordinate point at 50 - 200 point/second of rates (that is, 50 - 200 interrupts shall occur in 1 second). Here, the process which reads a pen coordinate is made into a transmitting process, and this read coordinate is transmitted to a receiving process.

[0082] (1) If a user puts a pen into the influence field of a tablet, an interrupt will occur and a transmitting process will acquire the pen information 2 for one coordinate of the 1st point. The transmitting process which obtained the coordinate data sends a signal to a receiving process, after registering into a message queue 21 the message which gains a data area from a message buffer 2, and writes in coordinate data and which reached and set up the priority highly. The coordinate data obtained by the time the 2nd point to the next user separated the pen from the influence field of a tablet is written in the exclusive buffer 19 one by one together with control Flagg"1." under the present circumstances – if control Flagg"1" for one point and coordinate data are written in – next control Flagg – "0" – it clears and the notation of an end is written in.

[0083] (2) The receiving process which received the signal from (1) takes out a message with the highest priority out of the message which has accumulated in the message queue 21. After reading data (coordinate data) from the applicable data area of a message buffer 2 based on this message and processing coordinate data, the data after the 2nd point of control Flagg"1" (coordinate data) are read from the exclusive buffer 19 one by one. Under the present circumstances, control Flagg is not "1", and when not yet written in, it waits for a while (for example, since the data for one coordinate will be obtained in about 0.02 seconds when the rate of a pen is carried out in 50 points/second, it waits for 0.02 seconds). Data are again read from the exclusive buffer 19. Even if it waits fixed time amount (for example, 0.5 seconds or more) continuously, in not becoming control Flagg"1", a pen judges it as what is distant from the influence field of a tablet, and it completes it.

[0084] By the above, when inputting coordinate data from a tablet using a pen, it goes message buffer 2, a priority is made high, it transmits to a receiving process, and the first coordinate data of the 1st point transmits coordinate data to a high speed via the exclusive buffer 19 from between the 2nd point at a receiving process. By these, a special signal becomes nothing possible [transmitting coordinate data to a receiving process from a transmitting process at a high speed].

[0085]

[Effect of the Invention] After setting a priority as a message highly and registering with a message queue 21 while writing data in the data area of a message buffer 2 as explained above, according to this invention, a signal in a receiving process Delivery, The receiving process which received the signal takes out a message with a high priority from a message queue 21. Read data from the data area in a message buffer 2 based on this, or Data are written in for the ability forming an exclusive buffer. Specially a signal in a receiving process Delivery, Since the configuration which receives a signal specially, a receiving process reads data from an exclusive buffer, or combines both, and transmits data to a receiving process from a transmitting process preferentially is adopted, Data required for the priority processing generated while the communication link was carried out to the receiving process from the transmitting process can be passed preferentially. While being able to process a communication link to process preferentially by preparing a priority in a message (data) especially, forming an exclusive buffer, or using both together at a high speed, the communication link of the data generated by very high frequency can be processed.

[Claim(s)]

[Claim 1] The message region acquisition section which gains a field from a message buffer (2) in the interprocess communication method which communicates between processes (3), The data write-in section which writes in the data which it is going to transmit to this gained field (4), The transmitting process which prepared the message queue registration section (5) which registers into a message queue (21) the message which set up the priority about data, It responds to having received the signal which this transmitting process transmitted. The message priority analysis section which finds a message with a high priority (or message most registered early when there was a message with two or more high priorities) out of the message registered into the message queue (21) (10), The interprocess communication method characterized by having the receiving process which prepared the data read-out section (12) which reads the data corresponding to a message with this found high priority from the field of the above-mentioned message buffer (2).

[Claim 2] The exclusive buffer which writes in data and control Flag who writes in and expresses settled in the interprocess communication method which communicates between processes (19), The transmitting process which prepared the data write-in section (15) which writes writing and above-mentioned control Flag in this exclusive buffer (19), and sets to it the data which it is going to transmit settled, The interprocess communication method characterized by having the receiving process which prepared the data read-out section (17) which control Flag in the above-mentioned exclusive buffer (19) writes in, and distinguishes whether it is settled for every predetermined time, and reads data when finishing [writing], and which reaches and resets control Flag.

[Claim 3] The exclusive buffer which writes in data in the interprocess communication method which communicates between processes (19), The transmitting process which prepared the data write-in section (25) which writes in the data which it is going to transmit to this exclusive buffer (19), The interprocess communication method characterized by having the receiving process which prepared the data read-out section (32) which reads data from the above-mentioned exclusive buffer (19) corresponding to

having received the special signal which this transmitting process transmitted.

[Claim 4] The interprocess communication method characterized by having both claim 1 and claim 3.

ibmf100275000jp07084804.doc